ATCOM® GSM Card AX-4G

Product Guide

Version: 1.2

Written by Robert.Ao

22-11-2010

Content

CONTACT ATCOM 2
CHAPTER 1 THE INTRODUCTION OF AX-4G 3
CHAPTER 2 HARDWARE INTRODUCTION 5
CHAPTER 3 SOFTWARE INSTALLATION 7
CHAPTER 4 SOFTWARE CONFIGURATION 9
CHAPTER 5 TEST 12
CHAPTER 6 SENDING SMS 14
CHAPTER 7 REFERENCE 15

Contact ATCOM

The Introduction of ATCOM

Founded in 1998, ATCOM technology has been always endeavoring in the R&D and manufacturing of the internet communication terminals. The product line of ATCOM includes IP Phone, USB Phone, IP PBX, VoIP gateway and Asterisk Card.

Contact sales:

Address	District C, east of 2nd floor, #3, Crown industry buildings, Chegongmiao Industry area, Futian district, Shenzhen, China
Tel	+(86)755-23487618
Fax	+(86)755-23485319
E-mail	sales@atcomemail.com

Contact Technical Support:

Tel	+(86)755-23481119
E-mail	Support@atcomemail.com

Website address: http://www.atcom.cn/

ATCOM Wiki Website: http://www.openippbx.org/index.php?title=Main_Page

Download Center: http://www.atcom.cn/download.html

Chapter 1 the Introduction of AX-4G

Overview of the AX-4G

AX-4G Asterisk card is the GSM channel PCI card that supports up to four GSM channels.
Using AX-4G GSM card, open source Asterisk PBX and stand alone PC, users can create their SOHO telephony solution which includes all the sophisticated features of traditional PBX, and extended features such as voicemail in IP PBX.

Features

4 GSM quadband channels(850/900/1800/1900MHz)
1 SIM card per GSM channel
DTMF detection
GSM data connections
GPRS data connections
Monitor GSM signal strength by command and LEDs
Monitor communication status by command line
Send SMS by Asterisk CLI, shell command and dail-plan
Detect accurately when the called party answers the call
Suitable for 3.3 volts and 5.0 volts 32 bit PCI slots
Pci slot or 4-pin connector power feeding

Applications

GSM connectivity for PBX
Mobile PBX
GSM VoIP gateway
SMS gateway
GSM callback services

Hardware Requirement

500-Mhz Pentium III
64MB RAM
3.3V or 5V PCI 2.2 slot

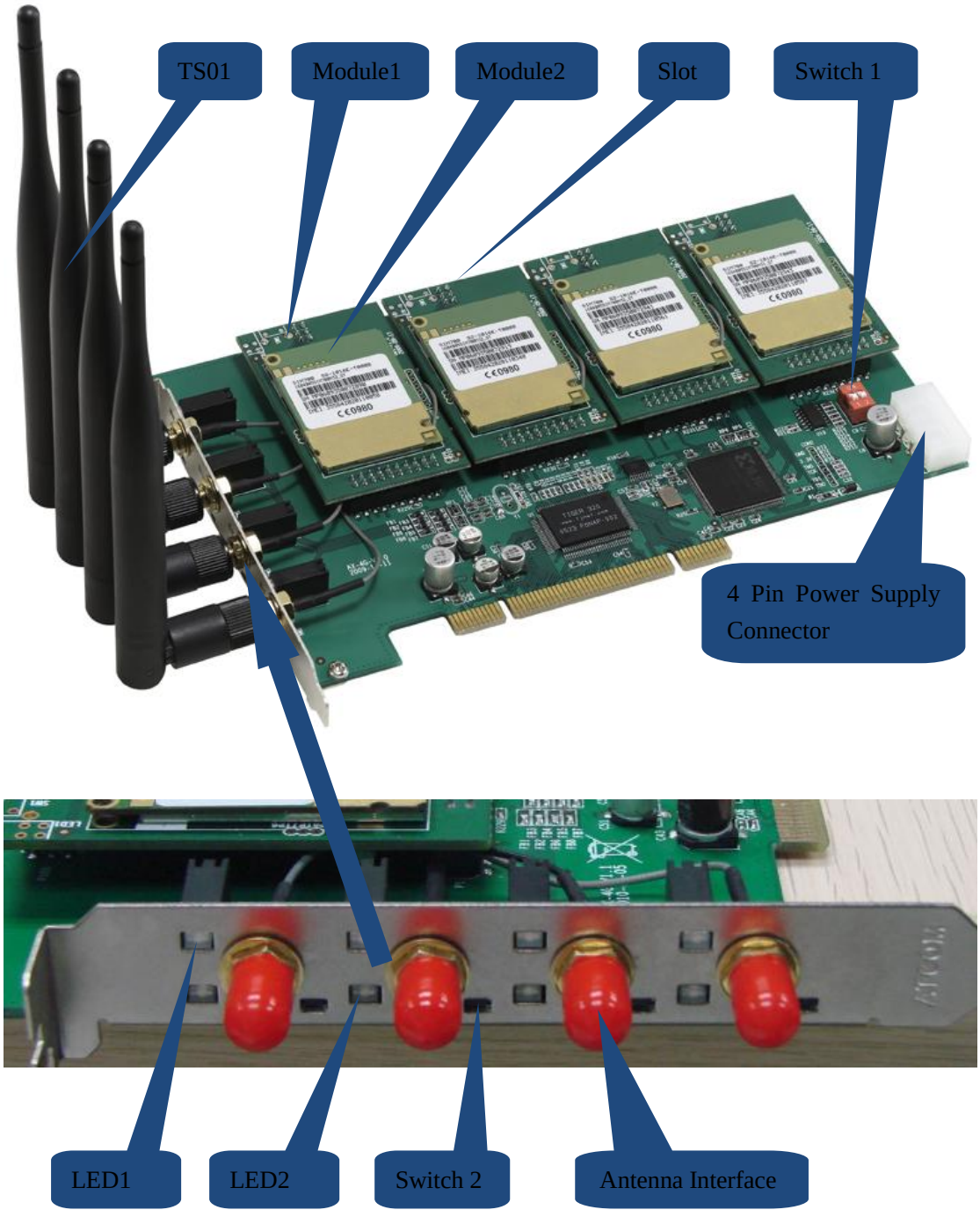
Measurement and Weight

Inner box	130 * 47 * 25mm
G.W./unit	0.25kg
Carton MEAS	545 * 370 * 230 mm
Units per Carton	40 units/ CTN
G.W./CTN	10 kg/CTN

Package

Item	Quantity
AX-4G mother board	1
GSM module (GX01)	1-4

Chapter 2 Hardware Introduction



TS01: Omnidirectional antenna

GX01: Module1+Module2

Module1: Module1 is used to load Module2, and one Module1 supports one cell phone card slot.

Module2: GSM/GPRS/EDGE 850/900/1800/1900MHz wireless module

Slot: SIM card of cell phone slot

Switch 1: There are two ways to provide power feeding for AX-4G, customers can use PCI slot to give power feeding directly, in this situation, please turn the switch1 to 1 on and 2 off; customer can also give power feeding through 4 pin power connector, and in this situation, please turn the swith1 to 1 off and 2 on.

4 Pin Power Supply: 4 pin power supply connector.

LED1: When the channel is in communication status, LED1 will be green steady.

LED2: Network status indicator of SIM card. When the SIM card is searching network, the LED2 will blink fast; when the SIM card has found the GSM network and registered successfully, the LED2 will flip at lower frequency.

Switch 2: Customer can turn off or turn on the SIM card manually through switch 2.

Antenna Interface: An interface used to connect with antenna.

Attention: In order to enhance the GSM signal in some hard environment, except the Antenna in the illustration above, ATCOM can also provide one other type of antenna(TL01) which has long cable(about one meter long) and can be connected to outside.

Chapter 3 Software Installation

Test Environment:

dahdi-linux-2.3.0 (download from ATCOM website)
dahdi-tools-2.3.0
asterisk-1.6.2.6 (download from ATCOM website)
centos 5.4 (kernel version: 2.6.18-164.el5)
AX-4G+4*GX01

After inserting the card into your PCI slot and boot your server, please use the "lspci" command to check the PCI bus compatibility. From the correct output, you can see the following line:

01:01.0 Communication controller: Tiger Jet Network Inc. Tiger3XX Modem/ISDN interface

---

A Tiger Jet device will be found, if you can not see the **Tiger Jet** device, please poweroff your server and try another PCI slot, if it still does not help, you have to check the compatibility issue between the card and your PCI bus.

1. To install asterisk and dahdi, we have to use "yum" command to install the following prerequisite packages:
   bison bison-devel zlib zlib-devel openssl openssl-devel gnutls-devel gcc gcc-c++

2. Download dahdi-linux, dahdi-tools, and asterisk
   Attention: please download dahdi-linux and asterisk from ATCOM website, because ATCOM has modified the source code for AX-4G.
   [root@localhost src]#
   wget http://www.atcom.cn/downloads/TelephonyCard/drivers/AX-4G/asterisk-1.6.2.6.tar.gz

   [root@localhost src]#
   wget
   http://www.atcom.cn/downloads/TelephonyCard/drivers/AX-4G/dahdi-linux-2.3.0.tar.gz

   [root@localhost src]#
   wget http://www.atcom.cn/downloads/TelephonyCard/drivers/AX-4G/dahdi-tools-2.3.0.tar.gz

3. Install dahdi-linux and dahdi-tools
   Install dahdi-linux
   1) [root@localhost src]# tar -xvzf dahdi-linux-2.3.0.tar.gz
   2) [root@localhost dahdi-linux-2.3.0]# make
   3) [root@localhost dahdi-linux-2.3.0]# make install

Install dahdi-tools

1) [root@localhost src]# tar -xvzf dahdi-tools-2.3.0.tar.gz
2) [root@localhost dahdi-tools-2.3.0]# ./configure
3) [root@localhost dahdi-tools-2.3.0]# make
4) [root@localhost dahdi-tools-2.3.0]# make install
5) [root@localhost dahdi-tools-2.3.0]# make config

4. Install asterisk

1) [root@localhost src]# tar -xvzf asterisk-1.6.2.6.tar.gz
2) [root@localhost asterisk-1.6.2.6]# ./configure
3) [root@localhost asterisk-1.6.2.6]# make
4) [root@localhost asterisk-1.6.2.6]# make install
5) [root@localhost asterisk-1.6.2.6]# make samples

Chapter 4 Software Configuration

1. In order to load the driver for ax4g once dahdi starts, please edit file:/etc/dahdi/modules
   Enter into /etc/dahdi/modules file, find the following two lines:
   # Digium TDM400P: up to 4 analog ports
   wctdm
   Then delete the wctdm, and add the following lines:
   ax4g
   ax400p
2. Please restart dahdi by running **/etc/init.d/dahdi restart** command.
3. Please check if the AX-4G driver has been loaded successfully, run cat /proc/interrupts command and you should see one line with ax4g:
   20: 2310670 2218516 IO-APIC-fasteoi ax4g
   If you can not see ax4g, you can try to poweroff your server and replug AX-4G into another PCI slot.
4. Please run the "dahdi_genconf" command to configure the /etc/dahdi/system.conf file and generate /etc/asterisk/dahdi-channels.conf file.
   [root@localhost ~]# dahdi_genconf
   It does not show any output if dahdi_genconf run successfully.

After running dahdi_genconf successfully, the system.conf file will get the following configuration:
/etc/dahdi/system.conf:

```
# Autogenerated by /usr/sbin/dahdi_genconf on Thu Jul  1 11:33:13 2010
# If you edit this file and execute /usr/sbin/dahdi_genconf again,
# your manual changes will be LOST.
# Dahdi Configuration File
# This file is parsed by the Dahdi Configurator, dahdi_cfg
# Span 1: WCTDM/4 "ATCOM AX4G Board 5" (MASTER)
fxsks=1
echocanceller=mg2,1
fxsks=2
echocanceller=mg2,2
fxsks=3
echocanceller=mg2,3
fxsks=4
echocanceller=mg2,4

# Global data
loadzone            = us
defaultzone         = us
```

After running dahdi_genconf successfully, the dahdi-channels.conf file will get the following configuration:

```
; Span 1: WCTDM/4 "ATCOM AX4G Board 5" (MASTER)
;;; line="1 WCTDM/4/0 FXSKS    (SWEC: MG2)"
signalling=fxs_ks
callerid=asreceived
group=0
context=from-pstn
channel => 1
callerid=
group=
context=default

;;; line="2 WCTDM/4/1 FXSKS    (SWEC: MG2)"
signalling=fxs_ks
callerid=asreceived
group=0
context=from-pstn
channel => 2
callerid=
group=
context=default

;;; line="3 WCTDM/4/2 FXSKS    (SWEC: MG2)"
signalling=fxs_ks
callerid=asreceived
group=0
context=from-pstn
channel => 3
callerid=
group=
context=default
;;; line="4 WCTDM/4/3 FXSKS    (SWEC: MG2)"
signalling=fxs_ks
callerid=asreceived
group=0
context=from-pstn
channel => 4
callerid=
group=
context=default
```

5. Please add the following line at the bottom of the chan_dahdi.conf file:
   #include dahdi-channels.conf

6. Please run asterisk with the following command:
   asterisk –vvgc
   reload

7. Please run dahdi show channels command
   You should see 4 channels as the following:
   *CLI> dahdi show channels

Chan	Extension	Context	Language	MOH Interpret	Blocked	State
pseudo		default		default		In Service
1		from-pstn		default		In Service
2		from-pstn		default		In Service
3		from-pstn		default		In Service
4		from-pstn		default		In Service

   *CLI>

Chapter 5 Test

1. Please check the status of the cell phone card plugged in the ax4g.
   Please type the following command three times, the result of the third time is correct:
   cat /proc/dahdi/ax4g_sta_card_1
   You can get one line with the following information:
   S0:4, S1:4, S2:4, S3:2
   The S0, S1, S2, S3 stand for the four modules on the ax4g orderly, and different numbers behind colon stand for different status of cell phone card.
   a: it means that the cell phone card is sending SMS.
   1: it means that the module2 is powered off.
   2: it means that the cell phone card is still searching for GSM network.
   3: it means that the cell phone card has registered successfully before, but now it failed.
   4: it means that the cell phone card has found GSM network successfully, and it is in idle status(not used by calling or sending SMS).
   5: it means that after a communication, one side hangs up the phone, but the other side has not hanged up.
   6: it means that the cell phone card is dialing-out.
   7: it means that somebody from outside is calling the channel, but no extension answers the phone, the extension is ringing.
   8: it means that an extension answers an incoming call, but the communication has not yet established, at this moment, the called party hangs up, then the caller do not need to be paid for this call, because this call has not been established really.
   9: it means that the channel is in communication status, and the call has been established really.

2. Please check the network signal of the cell phone card plugged in the ax4g, type the following command three times, the result of the third time is correct:
   cat /proc/dahdi/GSM_signal_card_1
   you can get one line with the following information:
   SQ0 = 31, SQ1 = 30, SQ2 = 18, SQ3 = 28
   The SQ0, SQ1, SQ2, SQ3 stand for the four modules on the ax4g orderly, and different numbers behind equality sign stand for the network signal of the cell phone card, the numbers 0 and 99 mean no signal, for other numbers, the bigger the number is, the stronger the signal is.

3. Check when the called party answers the call, when you dial-out through AX-4G.
   Please use **dmesg** command to detect if the called party answers the call when you dail-out through AX-4G, if the called party has answered the call, after running **dmesg** command, you can get one line with GX PR, X stands for the channel number. This feature will help customer to do billing accurately.

Edit a dial-plan, and use AX-4G to make outgoing calls and get incoming calls.

4. Please edit a dial-plan in extensions.conf file, you can refer to the following dial-plan:

```
[from-internal]
exten=>_1.,1,Dial(dahdi/1/${EXTEN:1})
exten=>_1.,2,Hangup()
exten=>_2.,1,Dial(dahdi/2/${EXTEN:1})
exten=>_2.,2,Hangup()
exten=>_3.,1,Dial(dahdi/3/${EXTEN:1})
exten=>_3.,2,Hangup()
exten=>_4.,1,Dial(dahdi/4/${EXTEN:1})
exten=>_4.,2,Hangup()

[from-pstn]
exten=>s,1,Answer()
exten=>s,2,Dial(sip/500)
exten=>s,3,hangup()
```

5. Use a registered sip phone to make an outgoing call, according to the dial-plan above, if you want to make the outgoing call through the first channel, you have to add a prefix "1" when you dial out, if you want to make the outgoing call through the second channel, you have to add a prefix "2". Here, the destination number is 15128731752, so dialing 115128731752, then you can get the following output from asterisk CLI:

```
-----------------------------------------------------------------------------------------------------------
 *CLI>      == Using SIP RTP CoS mark 5
   --Executing [115128731752@from-internal:1] Dial("SIP/500-00000003",
      "dahdi/1/15128731752") in new stack
      -- Called 1/15128731752
      -- DAHDI/1-1 answered SIP/500-00000003
-----------------------------------------------------------------------------------------------------------
```

The called party rings, and pick up the phone, so the call is successful.

6. Please use a phone to make an incoming call, here, the SIM card number for the first channel is 15889587467, using a cell phone to dial 15889587467, the sip 500 rings, and you can get the following output from asterisk CLI:

```
-----------------------------------------------------------------------------------------------------------
*CLI>          -- Starting simple switch on 'DAHDI/1-1'
[Jul    1 11:42:53] NOTICE[20678]: chan_dahdi.c:8734 ss_thread: Got event 18 (Ring Begin)...
       -- Executing [s@from-pstn:1] Answer("DAHDI/1-1", "") in new stack
       -- Executing [s@from-pstn:2] Dial("DAHDI/1-1", "sip/500") in new stack
    == Using SIP RTP CoS mark 5
       -- Called 500
       -- SIP/500-00000004 is ringing
       -- SIP/500-00000004 answered DAHDI/1-1
-----------------------------------------------------------------------------------------------------------
```

Customer can test other three channels in the same way.

Chapter 6 Sending SMS

There are three ways for users to send SMS by ATCOM ax4g.

1. Sending SMS by asterisk CLI command directly.

   sendsms dahdi/1 15889588569 Welcome to ATCOM

   Through the command above, you can send a message "Welcome to ATCOM" to 15889588569(a cell phone number) using the first channel. The right output like the following:

   *CLI> sendsms dahdi/1 15889588569 Welocome to ATCOM
   -- Launching Sendsms(15889588569 Welocome to ATCOM) on DAHDI/1-1
   -- Hungup 'DAHDI/1-1'

2. Sending SMS by shell command

   Install a shell script: send.c which can be downloaded from ATCOM website, customers can send a SMS by shell command, and do not have to send SMS under asterisk CLI.

   Download send.c file and compile it:

   /usr/src#wget http://www.atcom.cn/downloads/TelephonyCard/drivers/AX-4G/send.c
   /usr/src# gcc -O2 -o send send.c
   /usr/src# cp send /var/lib/asterisk/agi-bin
   /usr/src# cp send /usr/bin

   Using the command **send dahdi/1 15889588569 Welocome to ATCOM**, users can send a message "Welcome to ATCOM" to 15889588569(a cell phone number) using the first channel.

3. Sending SMS by editing a dial-plan

   Download send.c file and compile it:

   /usr/src# wget http://www.atcom.cn/downloads/TelephonyCard/drivers/AX-4G/send.c
   /usr/src# gcc -O2 -o send send.c
   /usr/src# cp send /var/lib/asterisk/agi-bin
   /usr/src# cp send /usr/bin

   Edit a dial-plan like the following in the file /etc/asterisk/extensions.conf

   [from-internal]
   exten=>888,1,agi(send,dahdi/2,15889588569,Welcome to ATCOM)
   exten=>888,2,hangup()

   Dialing 888, users can send a message "Welcome to ATCOM" to 15889588569(a cell phone number) using the second channel.

Chapter 7 Reference

http://www.asteriskguru.com/
http://www.asterisk.org/downloads
http://www.openippbx.org/index.php?title=Main_Page
http://www.atcom.cn/